

Schöne, Christian; Teaching Trends: Offen für neue Wege – Digitale Medien in der Hochschule (2014 : Oldenburg (Oldenburg))

Optimierung einer Lernumgebung für berufstätige Studierende. Ein Praxisbeispiel

Zawacki-Richter, Olaf [Hrsg.]; Kergel, David [Hrsg.]; Kleinfeld, Norbert [Hrsg.]; Muckel, Petra [Hrsg.]; Stöter, Joachim [Hrsg.]; Brinkmann, Katrin [Hrsg.]: Teaching Trends 2014. Offen für neue Wege: Digitale Medien in der Hochschule. Münster ; New York : Waxmann 2014, S. 171-187. - (Digitale Medien in der Hochschullehre; 2)



Quellenangabe/ Reference:

Schöne, Christian; Teaching Trends: Offen für neue Wege – Digitale Medien in der Hochschule (2014 : Oldenburg (Oldenburg)): Optimierung einer Lernumgebung für berufstätige Studierende. Ein Praxisbeispiel - In: Zawacki-Richter, Olaf [Hrsg.]; Kergel, David [Hrsg.]; Kleinfeld, Norbert [Hrsg.]; Muckel, Petra [Hrsg.]; Stöter, Joachim [Hrsg.]; Brinkmann, Katrin [Hrsg.]: Teaching Trends 2014. Offen für neue Wege: Digitale Medien in der Hochschule. Münster ; New York : Waxmann 2014, S. 171-187 - URN: urn:nbn:de:0111-pedocs-184689 - DOI: 10.25656/01:18468

<https://nbn-resolving.org/urn:nbn:de:0111-pedocs-184689>

<https://doi.org/10.25656/01:18468>

in Kooperation mit / in cooperation with:



WAXMANN
www.waxmann.com

<http://www.waxmann.com>

Nutzungsbedingungen

Gewährt wird ein nicht exklusives, nicht übertragbares, persönliches und beschränktes Recht auf Nutzung dieses Dokuments. Dieses Dokument ist ausschließlich für den persönlichen, nicht-kommerziellen Gebrauch bestimmt. Die Nutzung stellt keine Übertragung des Eigentumsrechts an diesem Dokument dar und gilt vorbehaltlich der folgenden Einschränkungen: Auf sämtlichen Kopien dieses Dokuments müssen alle Urheberrechtshinweise und sonstigen Hinweise auf gesetzlichen Schutz beibehalten werden. Sie dürfen dieses Dokument nicht in irgendeiner Weise abändern, noch dürfen Sie dieses Dokument für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, aufführen, vertreiben oder anderweitig nutzen. Mit der Verwendung dieses Dokuments erkennen Sie die Nutzungsbedingungen an.

Terms of use

We grant a non-exclusive, non-transferable, individual and limited right to using this document. This document is solely intended for your personal, non-commercial use. Use of this document does not include any transfer of property rights and it is conditional to the following limitations: All of the copies of this documents must retain all copyright information and other information regarding legal protection. You are not allowed to alter this document in any way, to copy it for public or commercial purposes, to exhibit the document in public, to perform, distribute or otherwise use the document in public.

By using this particular document, you accept the above-stated conditions of use.

Kontakt / Contact:

peDOCS
DIPF | Leibniz-Institut für Bildungsforschung und Bildungsinformation
Informationszentrum (IZ) Bildung
E-Mail: pedocs@dipf.de
Internet: www.pedocs.de



TEACHING TRENDS14

ELAN-e.V.-Kongress – Oldenburg

Offen für neue Wege:
Digitale Medien in der Hochschule

Olaf Zawacki-Richter
David Kergel
Norbert Kleinefeld
Petra Muckel
Joachim Stöter
Katrin Brinkmann
(Hrsg.)

DIGITALE MEDIEN
IN DER HOCHSCHULLEHRE
Eine Publikationsreihe des ELAN e.V.

herausgegeben vom

ELAN e.V.

Band 2

Olaf Zawacki-Richter, David Kergel,
Norbert Kleinefeld, Petra Muckel,
Joachim Stöter, Katrin Brinkmann
(Hrsg.)

Teaching Trends 2014

Offen für neue Wege:
Digitale Medien in der Hochschule



Waxmann 2014
Münster • New York



Gefördert durch



Niedersächsisches Ministerium
für Wissenschaft und Kultur



Bibliografische Informationen der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Digitale Medien in der Hochschullehre, Band 2

ISSN 2199-7667

ISBN 978-3-8309-3170-6

© Waxmann Verlag GmbH, Münster 2014

www.waxmann.com

info@waxmann.com

Umschlaggestaltung: Steffen Ottow, Clausthal-Zellerfeld

Titelbild: © kasto – Fotolia.com

Druck: Hubert und Co., Göttingen

Gedruckt auf alterungsbeständigem Papier, säurefrei gemäß ISO 9706



Printed in Germany

Alle Rechte vorbehalten. Nachdruck, auch auszugsweise, verboten.

Kein Teil dieses Werkes darf ohne schriftliche Genehmigung des Verlages in irgendeiner Form reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Inhalt

<i>Gabriele Heinen-Kljajić</i> Vorwort	7
<i>Claus R. Rollinger</i> Vorwort	9
<i>Olaf Zawacki-Richter</i> Vorwort	11
A: Forschendes Lernen mit digitalen Medien	
<i>Petra Muckel & David Kergel</i> Einführung: Forschendes Lernen mit digitalen Medien	13
<i>Sandra Hofhues, Gabi Reinmann & Mandy Schiefner-Rohs</i> Lernen und Medienhandeln im Format der Forschung	19
<i>David Kergel</i> Forschendes Lernen 2.0 – lerntheoretische Fundierung und Good Practice	37
<i>Birte Heidkamp</i> E-Science und forschendes Lernen	51
<i>Gerd Hoffmann</i> Hinzulernen im Verlauf des Forschenden Lernens auf Basis von Open Educational Resources. Unterstützung einer flexiblen Wissensvermittlung mit Referatorien	69
<i>Eva Poxleitner & Marlen Arnold</i> Forschungsbasiertes Lernen mit selbsterstellten Lernapps	83
B: Digitale Medien für heterogene Zielgruppen	
<i>Joachim Stöter, Svenja Bendenlier & Katrin Brinkmann</i> Einführung: Digitale Medien für heterogene Zielgruppen	99
<i>Carmen Schmitz-Feldhaus</i> Nicht traditionelle Studierende vs. traditionelle Studierende. Eine Onlinebefragung zum Sense of Coherence im Studium mit Fokus auf Menschen mit Beeinträchtigungen und Neue Medien	103
<i>Barbara Meissner & Hans-Jürgen Stenger</i> Agiles Lernen mit Just-in-Time-Teaching. Adaptive Lehre vor dem Hintergrund von Konstruktivismus und intrinsischer Motivation	121

Daniel Otto

Studentischer Austausch in der Fernlehre? A digital story!137

Karin Julia Rott

Medienkompetenz im Studium: Wie gut ist die Vorbereitung für das
spätere Berufsfeld?153

Christian Schöne

Optimierung einer Lernumgebung für berufstätige Studierende.
Ein Praxisbeispiel171

C: Bildungstechnologie und Medienkompetenz

Norbert Kleinefeld

Einführung: Bildungstechnologien und Medienkompetenz189

Rainer Jacob

Interaktive Whiteboards – der Einzug einer neuen Lerntechnologie.
Herausforderungen für die Schulen und für die Lehramtsausbildung.....191

Christian Greweling, Rüdiger Rolf & Denis Meyer

Automatisierte Vorlesungsaufzeichnungen mit Opencast Matterhorn
an der Universität Osnabrück. Wissenswertes zum praktischen Einsatz
des Systems, die technische Infrastruktur und mögliche Fallstricke.....203

Lisa Rupp, Benjamin Wulff & Kai-Christoph Hamborg

Veranstaltungsaufzeichnungen mit LectureSight: Effekte auf Lernen
und Akzeptanz217

Jana Riedel, Claudia Albrecht & Lars Schlenker

Die Didaktik zählt: Kompetenzvermittlung zur Lösung didaktischer
Herausforderungen233

*Stephan Tjettmers, Majana Beckmann, Marc Krüger, Ralf Steffen, Susanne Dräger,
Rüdiger Rhein & Oliver J. Bott*

Professionalisierung der Beratung zum Einsatz digitaler Medien in der Lehre.
Das Weiterbildungskonzept „Hochschuldidaktische Beratung“249

Autorinnen und Autoren265

Christian Schöne

Optimierung einer Lernumgebung für berufstätige Studierende

Ein Praxisbeispiel

Abstract

Wenn heterogene Zielgruppen für Online- oder Blended-Learning-Angebote gewonnen werden und sie diese erfolgreich absolvieren sollen, kommt es entscheidend darauf an, auch die Online-Lernumgebung an die Bedürfnisse dieser Zielgruppen anzupassen. Dies betrifft z. B. die Auswahl der unterstützten Endgeräte, die Benutzerführung, den Mix aus synchronen und asynchronen Elementen, die Auswahl der eingesetzten Tools und – nicht zuletzt – auch das Design der Lernumgebung.

Im Center für Lebenslanges Lernen (C3L) an der Universität Oldenburg wird seit 2007 eine eigens entwickelte internetgestützte Lernumgebung verwendet. Das Design bildet die Struktur von Studiengängen und Modulen über ein Zwei-Ebenen-Konzept in intuitiv verständlicher Weise ab und bietet Studierenden und Lehrenden einen schnellen Überblick aktueller Neuigkeiten.

In 2012 wurde beschlossen, das System von Grund auf neu zu entwickeln. Ein wichtiger Grund dafür war der Wunsch, die bisherige Plattform (Lotus Domino) auf eine Basis von Open-Source-Produkten umzustellen. Zugleich sollte das Potenzial aktueller Web-Technologien wie HTML5 und Echtzeit-/Push-Technologien für verbesserte Bedienbarkeit und Nutzerinteraktion genutzt werden.

Am Beispiel der neuen Lernumgebung des C3L wird gezeigt, wie eine solche Anpassung an das Profil der Zielgruppe (hier: berufstätige Studierende) gelingen kann. Dabei werden Potentiale neuer Web-Technologien für internetgestützte Lernumgebungen erläutert und es wird auf die Unterstützung mobiler Endgeräte eingegangen.

1 Zielgruppe, Blended-Learning-Szenario und didaktischer Grundrahmen

Die vom C3L betreuten berufsbegleitenden Studiengänge (BBS) der Universität Oldenburg sind speziell auf Personen mit eingeschränktem Zeitbudget ausgerichtet. Die Studierenden besitzen häufig mehrjährige Berufserfahrung oder unterliegen anderen zeitlichen Einschränkungen, etwa durch familiäre Verpflichtungen, und sind in der Regel etwas älter

als Studierende in den grundständigen Studiengängen. Das Durchschnittsalter liegt zwischen 25-30 Jahren (Bachelor) und zwischen 35-40 Jahren (Master) je nach Studiengang.

Um ein zeitlich flexibles Studieren zu ermöglichen, wird ein typisches Blended-Learning-Konzept mit alternierenden Online- und Präsenzphasen (vgl. Kerres, 2013)¹ angewendet. Ein Modul besteht im Regelfall aus fünf Phasen:

Die **Vorbereitungsphase** dient zur Bearbeitung der Studienmaterialien in eigener Regie sowie deren Diskussion in der Lernumgebung und beinhaltet in der Regel Online-Aufgaben zur Vertiefung der Inhalte. Die Vorbereitungsphase dauert ein bis zwei Monate. Im Anschluss findet die **erste Präsenzphase** statt. Der typische Zeitumfang beträgt zwei Tage; sie dient der Vertiefung einzelner Themen und der Bildung von Projektgruppen, die in der **Projektphase** jeweils ein Thema über ca. zwei Monate hinweg bearbeiten. Die Ergebnisse der Projektphase werden in einer Präsentation zusammengestellt und während der **zweiten Präsenzphase** (ebenfalls zwei Tage) vorgestellt und gemeinsam diskutiert. In der **Abschlussphase** besteht ggf. Zeit zur Überarbeitung von Projektergebnissen und abschließender Diskussion der Lerninhalte. Die Abschlussphase beträgt im Regelfall vier Wochen.

Der Schwerpunkt ist somit deutlich im Bereich E-Learning verortet. Die Online-Phasen werden durch Mentoren und Dozenten proaktiv betreut, die Kommunikation wird über die C3L-Lernumgebung abgewickelt. Um die Vermittlung von komplexem Wissen zu fördern, liegt den BBS ein konstruktivistisch, kompetenzorientiertes didaktisches Modell zu Grunde, welches wesentlich durch die Projektarbeit gestützt wird. Die Themenvorschläge für die Projektarbeit entstammen in der Regel von den Studierenden selbst und stehen daher häufig in Bezug zu den beruflichen Tätigkeiten der Teilnehmenden.

Eine Lernumgebung sollte in untergeordneter Weise dem Instruktionsdesign angepasst sein. Nach Schulmeister kann E-Learning in zwei Gruppen aufgeteilt werden: „Während der Studierende im E-Learning-Typ A sich vorwiegend mit vorgefertigten Lernobjekten auseinandersetzt, erarbeitet der Studierende im E-Learning-Typ B Wissen gemeinsam mit anderen“ (Schulmeister, 2005, S. 487). Die beiden genannten Typen von E-Learning können leicht mit zwei didaktischen Paradigmen assoziiert werden; während Typ A dem Instruktionsparadigma entspricht, folgt Typ B dem Konstruktionsparadigma (vgl. Zawacki-Richter, 2013).

Es ist naheliegend, dass die von Schulmeister vorgenommene Typisierung zu unterschiedlichen Qualitätsdimensionen für beide Typen von E-Learning führen muss. Er subsummiert die Unterschiede im Qualitätsbegriff folgendermaßen: „Während im Typ A die Qualität des Lernens überwiegend von der Interaktivität der Lernobjekte abhängt, beruht

1 Anmerkung: Kerres konnotiert den Begriff „Blended-Learning“ eher negativ. Für ihn drückt der Begriff Lernszenarien aus, in denen „Teile des konventionellen Unterrichts ersetzt werden durch mediengestützte Verfahren, der Rest des Unterrichts aber wie bisher stattfindet“ (Kerres, 2013, S. 410). Lernszenarien, in deren Planung der Einsatz unterschiedlicher Medien a priori berücksichtigt wird, bezeichnet er als „Hybride Lernarrangements“ (Kerres, 2013, S. 412). In diesem Text drückt Blended-Learning jedoch ein ganzheitliches Konzept aus, welches eher mit Kerres „Hybridem Lernarrangement“ übereinstimmt.

die Qualität der Lernprozesse im Typ B sehr auf der Qualität der Moderation“ (Schulmeister, 2005, S. 487).

Die wichtigste Aufgabe der C3L-Lernumgebung besteht somit darin, die bestmögliche Unterstützung für den Moderationsprozess zu bieten. Im Vordergrund stehen daher Werkzeuge für synchrone und asynchrone Kommunikation, wobei Mittel der asynchronen Kommunikation zwingend verfügbar sein müssen, um die zeitliche Flexibilität zu erhalten.

2 Medien-/Endgerätenutzung

Eine gute Übersicht zur Medien- und Endgerätenutzung durch Studierende bietet der ECAR-Report (Dahlstrom, 2013). Für die Entwicklung einer internetgestützten Lernumgebung ist vorrangig die Endgerätenutzung von Interesse. Die Bereitstellung von Studienmaterialien in verschiedenen Formaten betrifft das Design einer Lernumgebung nur indirekt und unter bestimmten Voraussetzungen. Besondere Gründe könnten beispielsweise multimedial aufbereitete, mit interaktiven Simulationen angereicherte Lerninhalte sein. Eine derartige Entscheidung gehört jedoch zu einem übergeordneten Bereich des Instruktionsdesigns – interaktive Simulationen sind beispielsweise besonders für Unterricht nach dem Instruktionsparadigma (vgl. Kapitel 1) von Bedeutung.

2.1 ECAR-Studie

An der aktuellen ECAR-Studie nahmen N=113.000 Studierende von insgesamt 251 Institutionen aus 14 Ländern teil, die überwiegende Mehrheit bilden Studierende aus den USA.

Abbildung 1 bietet eine Übersicht zum Besitz von Endgeräten der befragten Studierenden. Die erste Erkenntnis ist, dass Studierende nahezu ausnahmslos über einen Laptop, Desktop- oder Tablet-PC verfügen. Weiterhin ist klar ersichtlich, dass der Trend in Richtung mobiler Geräte zeigt: Tablets, Laptops und Smartphones ersetzen zunehmend den klassischen Desktop-PC. Interessant ist auch, dass die Verbreitung dedizierter E-Book-Reader im Vergleich dazu deutlich langsamer voranschreitet. Vermutlich ist dies dem Umstand geschuldet, dass E-Books problemlos mit Smartphones, Laptops und Tablet-PCs gelesen werden können, während E-Reader spezialisierte Geräte sind.

Neben dem Besitz von Endgeräten wurden die Teilnehmenden der ECAR-Studie auch zu Einsatz und Nutzen unterschiedlicher Endgeräte für das Studium befragt. Die Ergebnisse sind in Abbildung 2 dargestellt. Das wichtigste technische Werkzeug von Studierenden ist aktuell eindeutig der Laptop mit steigender Nutzung und Bedeutung. Die Nutzung für akademische Zwecke ist bei allen dargestellten Endgeräten gestiegen, die Bedeutung ist aber lediglich bei Smartphones und Laptops merklich angestiegen. Internetgestützte Lernumgebungen sollten daher im Schwerpunkt für die Nutzung durch PCs optimiert sein – rein technisch muss dabei zwischen Laptops und Desktop-PCs nicht unterschieden werden. An zweiter Stelle erscheint es sinnvoll Unterstützung für Smartphones zu bieten, bzw. diese auszubauen. Die Zukunft von Tablet-PCs im akademischen Bereich ist aktuell schwer einzuschätzen. Die Nutzung ist zwar stark angestiegen, die Bedeutung jedoch

nicht. Möglicherweise führt der enorme Anstieg in der Verbreitung von Tablets (vgl. Abbildung 1) schlicht dazu, dass Studierende häufiger ein Tablet im Gepäck haben und dieses dann ggf. auch dafür nutzen, um Informationen im Rahmen des Studiums abzurufen.

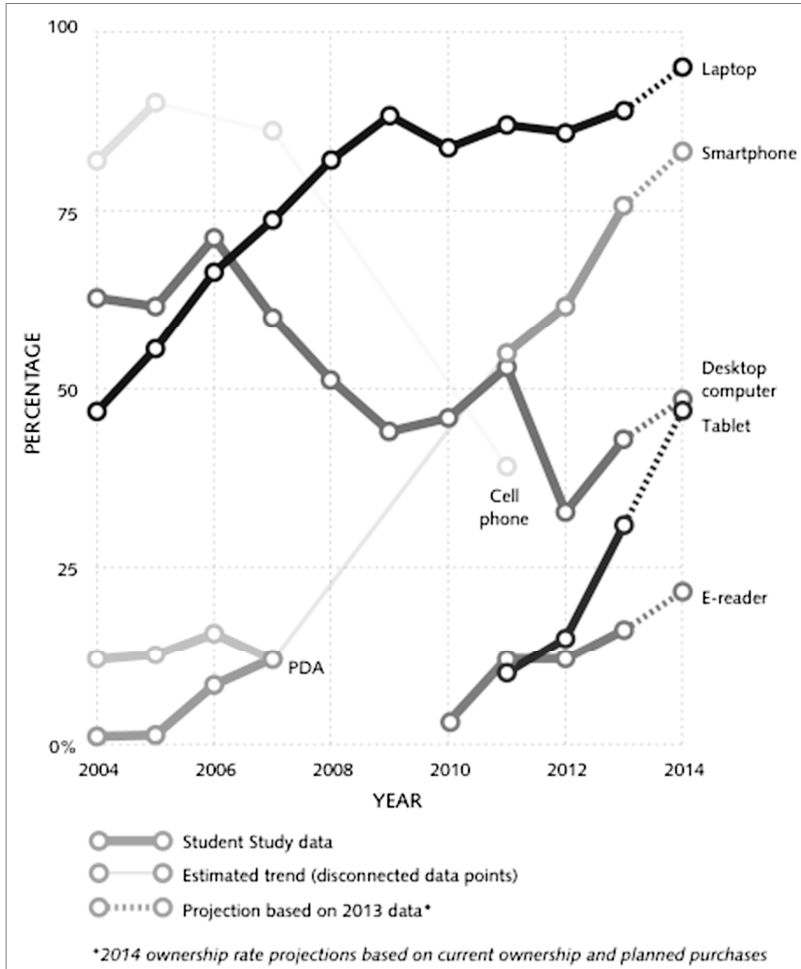


Abbildung 1: Besitz von Endgeräten (Dahlstrom, 2013, S. 26)

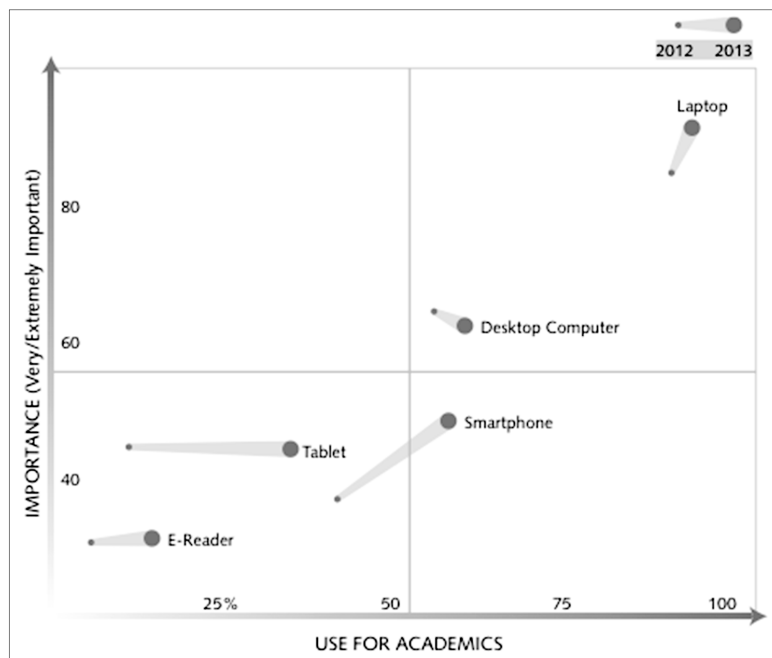


Abbildung 2: Nutzung von Endgeräten im Studium und relative Wichtigkeit (Dahlstrom, 2013, S. 27)

2.2 Studie „Mediennutzung im Studium“

Im Rahmen des vom BMBF geförderten Projektes MINT-Online² wurde eine Studie zur Mediennutzung im Studium durchgeführt (Zawacki-Richter, Hohlfeld & Müskens, 2014). Die Teilnehmerzahl ist mit $N=2.338$ um zwei Größenordnungen kleiner als die aktuelle ECAR-Studie, bietet aber eine gezielte Differenzierung zwischen ‚traditionellen Studierenden‘ (TS) ($N_{TS}=1.508$) und ‚nicht traditionellen Studierenden‘ (NTS) ($N_{NTS}=771$). Die Kriterien für die Zuordnung zur Gruppe der NTS sind vergleichbar mit der Zielgruppendefinition der BBS. Konkret wurden Teilnehmende der Gruppe der NTS zugeordnet, wenn sie:

- An einem Fern-/Onlinestudium teilnehmen,
- oder in Teilzeit studieren,
- oder mehr als 19 Stunden/Woche berufstätig,
- oder älter als 30 Jahre sind.

Der explizite Zweck dieser Studie ist es, „Aufschluss über die formelle und informelle Medien- und Lernumwelt der Studierenden, die Implikationen für die Gestaltung und

² www.mintonline.de (08.07.2014).

Weiterentwicklung medialer Lernumgebungen“ (Zawacki-Richter, Hohlfeld & Müskens, 2014, S. 0) zu erhalten.

In Abbildung 3 sind die Ergebnisse der Studie für den Besitz von Endgeräten dargestellt. Rein qualitativ unterscheiden sich (die vergleichbareren) Ergebnisse kaum, lediglich der Anteil Studierender im Besitz mobiler Endgeräte fällt hier deutlich geringer aus als in der ECAR-Studie.

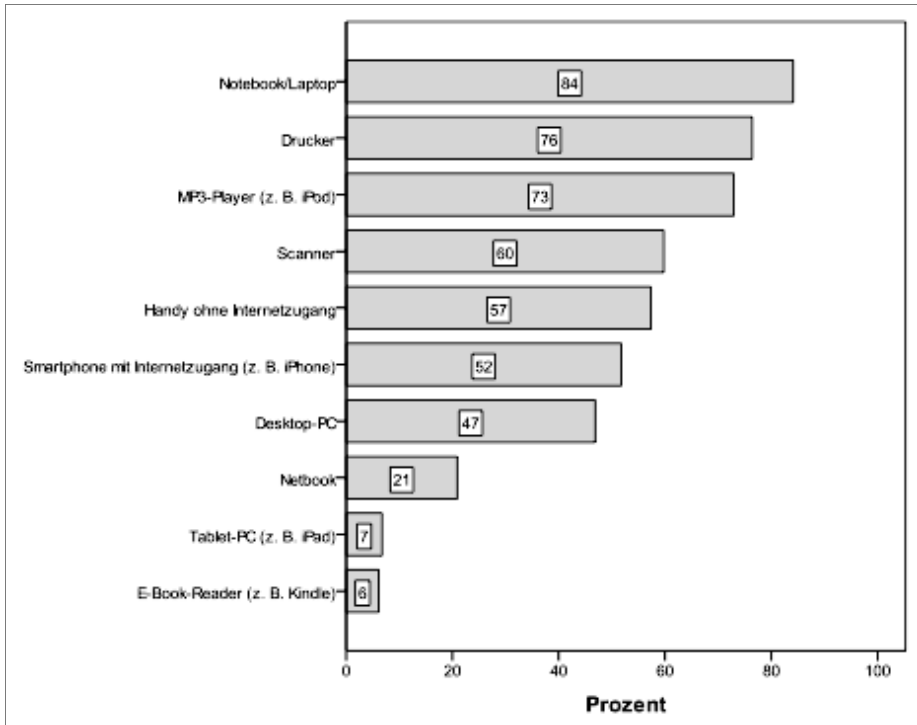


Abbildung 3: Besitz von Endgeräten durch Studierende (N=2.338) (Zawacki-Richter, Hohlfeld & Müskens, 2014, S. 10)

Es ist davon auszugehen, dass dies lediglich in dem traditionell etwas konservativerem Kaufverhalten deutscher Konsumenten gegenüber amerikanischen begründet ist und wird sich sehr wahrscheinlich in den nächsten Jahren angleichen. Interessant sind die Ergebnisse der (in der Abbildung nicht aufgeführten) Unterschiede zwischen traditionellen und nicht traditionellen Studierenden: Der Studie nach besitzen nicht traditionelle Studierende häufiger „ein Tablet-PC (z. B. iPad) und E-Book-Reader (z. B. Kindle): 11% gegenüber 6% bzw. 11% gegenüber 4%. Beim Smartphone mit Internetzugang (z. B. iPhone) ist es jedoch umgekehrt: 54% der traditionellen Studierenden gegenüber 50% der NTS besitzen ein Smartphone. In allen anderen Bereichen sind die Verhältnisse relativ ausgeglichen“ (Zawacki-Richter, Hohlfeld & Müskens, 2014, S. 10).

Insgesamt zeigt die Studie auf, dass nicht traditionell Studierende ein breiteres Spektrum an Medien und technischen Hilfsmitteln einsetzen und sich zudem weitere technische und mediale Unterstützung wünschen. Im Fazit der Studie wird hervorgehoben, dass Lernumgebungen insbesondere für berufstätige Studierende Werkzeuge für kooperatives Lernen und Arbeiten anbieten und mit mobilen Endgeräten nutzbar sein sollten (Zawacki-Richter, Hohlfeld & Müskens, 2014, S. 31f.).

3 C3L-Lernumgebung

Das Center für Lebenslanges Lernen betreibt seit 2007 eine eigens entwickelte Lernumgebung auf der Basis von IBM (Lotus) Domino. Die wichtigsten Gründe für eine eigene Lernumgebung sind:

- Reduktion auf die benötigten Funktionen ermöglicht minimale Komplexität und intuitiv einfache Nutzung;
- die Unabhängigkeit von externen Dienstleistern erleichtert Anpassungen und Neuerungen;
- direkte intra-institutionelle Kommunikation zwischen didaktischem Personal und Entwicklern der Lernumgebung.

Dabei ist der letzte Punkt nicht zu unterschätzen – in der Regel kommunizieren Programmentwickler/innen von und Lehrende in Studiengängen mit ihrer IT-Abteilung über Einsatz, Konfiguration und Entwicklung von Lernplattformen. Bekanntermaßen haben beide Personenkreise häufig unterschiedliche Ansichten. Ohne dass ein ernsthaftes Konfliktverhältnis vorliegt, sind Schwierigkeiten in der Kommunikation oftmals vorhanden, zudem muss die „institutionelle Firewall“ überwunden werden. Größere strukturell/organisatorische und persönliche Nähe erleichtern die Kommunikation zwischen Expert/inn/en für Didaktik und IT in erheblichem Maße.

Nach mehr als fünf Jahren des erfolgreichen Betriebs der C3L-Lernumgebung wurde 2012 beschlossen, die Lernumgebung mit Ruby on Rails (Hansson, kein Datum) neu zu implementieren. Diese Entscheidung wurde durch ausstehende Modernisierungen und den Wunsch eine andere technische Basis zu verwenden motiviert.

Ruby on Rails ist ein quelloffenes Web-Applikation Framework, dass in der Programmiersprache Ruby (Ruby-Community, kein Datum) verfasst ist und in den letzten Jahren zunehmend große Verbreitung gefunden hat. Das Framework setzt auf eine Model-View-Controller-Architektur (MVC) zur logischen Trennung zwischen Datenmodell, Kontrollfluss und Darstellung. Die Verwendung der REST-Architektur (Representational State Transfer) stellt einen Aspekt der ‚convention over configuration‘-Philosophie von Rails dar, welche insgesamt eine agile Entwicklung und Rapid-Prototyping erleichtert. Zudem existieren zahlreiche frei verfügbare und ausgereifte Bibliotheken für Ruby und RoR, über die viel Entwicklungsaufwand eingespart werden kann.

Die Entwicklungsarbeiten wurden im Rahmen des Projektes Mint.online durchgeführt.

3.1 Übersicht C3LLO

Ein Grundkonzept der C3L-Lernumgebung ‚C3LLO‘ ist die Verwendung von zwei inhaltlichen Ebenen zur Darstellung von Modulen und des Studiengangs. Der Studiengang wird als Hauptebene abgebildet und bietet neben auswählbaren Funktionen eine Übersicht neuer Ereignisse, Termine, aktueller Mitteilungen und belegten Modulen auf der Startseite. Die Startseite eines Moduls bietet ebenfalls eine Übersicht der modulspezifischen Neuigkeiten sowie eine Schnellübersicht der Arbeitsgruppen. Studierende können so umgehend auf die Funktionen ihrer Gruppe(n) zugreifen und Lehrende haben eine schnelle Übersicht aller Gruppen. Zwischen der Haupt- und Modulebene kann über einen zentralen Reiter jederzeit gewechselt werden. Der Wechsel zwischen Modulen oder ggf. auch Studiengängen ist jederzeit über eine Sprungnavigation möglich. Diese intuitiv einfache Struktur wurde bei der Neuentwicklung übernommen.

Alle Funktionen der Lernumgebung (wie etwa Foren, Dateiablagen, Nachrichtenfunktion u.v.m.) können beliebig einer der Ebenen oder einer Gruppe zugeordnet werden. Diese Kopplung wird durch eine sehr einfache Schnittstelle realisiert, in der lediglich die Zuordnung zu einem Personenkreis (Modul, Studiengang, Gruppe) und die entsprechende Verankerung im Navigationsmenü definiert werden. Durch die einfache Schnittstelle können neue Funktionen auf einfache Weise hinzugefügt werden, die lose Art der Kopplung minimiert zudem die, bei Plugin-Systemen häufig vorhandenen, Einschränkungen und vorausgesetzte Kenntnis des Gesamtsystems. Funktionen werden vom System anhand der implementierten Schnittstelle automatisch erkannt und sind unmittelbar nach ihrer Erstellung verfügbar.

Bei der Einrichtung von Studiengängen und Modulen werden Funktionen und Menüstrukturen entweder per Drag&Drop erstellt oder aus benutzerdefinierten Templates erzeugt (dies beinhaltet die Vorauswahl von Funktionen für Gruppen). Die Funktionen Forum, Nachrichten/Mail und Dateiablage bilden die Grundlage einer jeden Lernumgebung. Aus diesem Grund wurde hier besonderer Wert auf intuitive Bedienung und Übersichtlichkeit gelegt. Postfächer werden beispielsweise mit sortierbaren Tabellen dargestellt, um Nachrichten schnell nach Eingangsdatum, Betreff, Sender bzw. Empfänger oder der Anzahl der Anhänge zu sortieren. Die Dateiablage unterstützt Drag&Drop-Interaktionen zum Verschieben von Dateien und Ordern (vgl. Kapitel 3.3.1 Dateiablage mit Drag&Drop-Upload), das Forum bietet die Möglichkeit Entwürfe zu erstellen, bei der Darstellung eines Beitrags kann jederzeit der aktuelle Thread eingeblendet und darin navigiert werden.

In der alten C3L-Lernumgebung wurden in den Projektgruppen Wikis bereitgestellt, um Unterstützung für die gemeinsame Entwicklung von Texten zu bieten. Diese wurden jedoch nur spärlich verwendet. Ein häufig genannter Grund dafür war die Notwendigkeit die Wiki-Syntax zumindest in den Grundzügen lernen zu müssen. Als Alternative bieten wir in C3LLO nun Etherpads an (vgl. Etherpad, Chat und Online-Awareness-Funktion in C3LLO) und evaluieren deren Nutzen für die Studierenden.

3.2 Unterstützung für mobile Endgeräte

Den aktuellen Erkenntnissen nach (vgl. Kapitel 2) sollte die Lernumgebung mit mobilen Endgeräten gut nutzbar sein. Bei der heutigen Anzahl verschiedener internetfähiger mobiler Endgeräte ist es nicht leicht, einen optimalen Zugang für alle Geräte zu realisieren. Es wurde daher entschieden, eine gesonderte Lösung für Smartphones zu entwickeln. Mit den meisten Tablet-PCs kann C3LLO über die reguläre, für Laptop und Desktop-PCs optimierte Web-Schnittstelle problemlos genutzt werden.

Zur Optimierung von Anwendungen für Mobiltelefone bieten sich zwei grundlegende Lösungsmöglichkeiten an:

- Native App,
- Stylesheet für kleine Bildschirme.

Beide Möglichkeiten haben verschiedene Vor- und Nachteile: Native Apps müssen für jedes Betriebssystem eigens entwickelt werden – aktuell müssen aufgrund erheblicher Marktpresenz mindestens zwei Systeme unterstützt werden: iOS (Apple) und Android (Google). Im Gegenzug bieten native Apps eine Reihe von Vorteilen:

- Zugriff auf Funktionen und Sensoren des Endgeräts;
- Designelemente und clientseitige Skripte sind Bestandteil der App, keine Übertragung von HTML, CSS, Javascript und Bildern notwendig;
- serverseitige Templates erfordern keine zusätzlichen Verzweigungen, um mobile Geräte gezielt zu berücksichtigen;
- häufig ist nur eine Teilmenge der Funktionen einer Webanwendung für mobile Nutzung sinnvoll.

Insbesondere für komplexere Webseiten (wie etwa Lernumgebungen) kann es daher durchaus sinnvoll sein, nativen Apps den Vorzug zu gewähren. Aus diesen Gründen sowie dem Umstand, dass Mobile-Learning aktueller Forschungsgegenstand im Projekt Mint.online ist, wurde eine App basierte Lösung gewählt. Die App wird aktuell für Android entwickelt, eine Portierung für iOS ist geplant.

3.3 Einsatz aktueller (Web-)Technologien

Die kommerzielle Nutzung des Internet ist im vergangenen Jahrzehnt rasant gestiegen. Diese Kommerzialisierung hat nicht zuletzt auch die Entwicklung neuer (Web-)Technologien vorangetrieben.

Ein kurzer Blick auf die Geschichte technischer Neuerungen im Bereich der Web-Technologien zeigt eine klare Verdichtung ab ca. 2008 auf.³ Der aktuelle HTML-Standard des W3C (W3C, 2014), besser bekannt als „HTML5“ ist der Versuch viele dieser Technologien in gebündelter Form zu standardisieren. Seit ein paar Jahren bieten die meisten be-

3 Eine schnelle Übersicht bietet hier etwa www.evolutionoftheweb.com (07.08.2014).

kannten Browser gute Unterstützung für große Teile des HTML5-Standards. Insbesondere ist es sehr erfreulich für Entwickler, dass Microsofts Internet Explorer ab Version 9 gute Unterstützung für W3C Standards bietet. Dadurch entfällt an zahlreichen Stellen die Notwendigkeit der doppelten Implementierung von Funktionen. Als Anregung zur Nutzung neuer Technologien werden im Folgenden zwei Komponenten der neuen Lernumgebung des C3L dargestellt: Eine Dateiablage mit Drag&Drop-Interaktion und der Einsatz von Push-Technologien.

3.3.1 Dateiablage mit Drag&Drop-Upload

Das Zusammenspiel von XMLHttpRequest Level 2 (W3C, 2012), Drag&Drop und der File API (W3C, 2013) eröffnet die interessante Möglichkeit einzelne oder mehrere Dateien asynchron hochzuladen. Durch die Verwendung von Drag&Drop entfällt die insbesondere beim Hochladen mehrerer Dateien lästige Auswahl jeder einzelnen Datei. XMLHttpRequest spezifiziert ein Objekt in Javascript, das Methoden bereitstellt, um HTTP-Anfragen durchzuführen. Dadurch können weitere Daten zwischen Browser und Webserver ausgetauscht werden, ohne die aktuelle Seite zu verlassen. Diese Technologie ist daher auch unter dem Namen „AJAX“ („Asynchronous Javascript and XML“) bekannt.

AJAX wurde erstmals 1999 in einem Browser integriert (als XMLHttpRequest ActiveX Steuerelement in Microsofts Internet Explorer 5), die erste W3C-Spezifikation des XMLHttpRequest-Objektes wurde 2006 veröffentlicht. Diese erlaubte jedoch nur den Transfer von zeichenbasierten Daten (z.B. Text, Html, XML, Javascript etc.). Version 2 bietet mitunter die Möglichkeit binäre Daten zu übertragen. Die „File API“ beschreibt einen Standard für Javascript-Objekte, die Dateien repräsentieren. Die wichtigsten Objekte sind das File-Objekt für eine einzelne Datei und das FileList-Objekt für eine Liste von Dateien.

Drag&Drop spezifiziert wiederum Javascript-Objekte und Ereignisse für das Verschieben von Elementen im – bzw. über dem Browserfenster. Beispielsweise können Dateien in einem Dateimanager markiert und in ein Browserfenster gezogen werden. Beim Ziehen über dem Browserfenster sowie beim Loslassen der Maustaste werden durch den Browser Ereignisse ausgelöst, die mit Javascript erfasst und verarbeitet werden können. Ohne gezielte Verarbeitung dieser Ereignisse führen viele Browser Standardoperationen durch, beispielsweise kann markierter Text in Eingabefelder oder die Adresszeile des Browsers gezogen werden.

Mit den oben kurz beschriebenen Schnittstellen ist es denkbar einfach, Dateien aus einem Dateimanager im Hintergrund auf den Webserver zu kopieren:

- „ondrop“-Ereignis für Drag-Interaktionen abfangen und prüfen, ob die Eigenschaft „dataTransfer“ des Ereignisses Dateien enthält. Dazu muss lediglich geprüft werden, ob das Feld „dataTransfer.files“ mindestens einen Eintrag (entspricht einer Datei) enthält.
- In „dataTransfer.files“ ggf. vorhandenen Dateien per XMLHttpRequest an den Server übertragen.

Der Fortschritt der Datenübertragung kann zudem leicht über das „onprogress“-Ereignis des XMLHttpRequest-Objektes überwacht und nach Belieben visualisiert werden. Ein einfacher Fortschrittsbalken kann leicht mit dem „Canvas“-Element realisiert werden. Dabei ist anzumerken, dass die Fortschrittsanzeige keine serverseitigen Skripte benötigt, um das bereits übertragene Datenvolumen zu ermitteln.

Der beschriebene Prozess inklusive Fortschrittsanzeige benötigt knapp 150 Zeilen Javascript (von insgesamt ca. 1.000 Zeilen Javascript für die gesamte Dateiablage).

3.3.2 Push-Technologien

Unter Push-Technologien versteht man Kommunikationsvorgänge, die nicht vom Empfänger initiiert wurden. Üblicherweise liegt Push-Kommunikation ein Subskriptionsmodell zugrunde, d.h. Empfänger/innen abonnieren die gewünschten Informationskanäle. Sobald neue Informationen verfügbar sind, werden diese den Empfänger/inne/n unaufgefordert übermittelt. Die Post betreibt in diesem Sinne einen Push-Dienst mit Briefkästen als Empfänger – die Zustellung bedarf keiner Aufforderung durch die Eigentümer/innen der Briefkästen. Das Leeren des eigenen Briefkastens hingegen entspricht im Gegensatz dazu einem Kommunikationsvorgang, den man selbst initiiert hat, und könnte als „Pull“- oder „Request“-Technologie bezeichnet werden. Regelmäßiges Abfragen von Informationen („Leeren des eigenen Briefkastens“) wird im IT-Jargon auch „Polling“ genannt.

Push-Technologien in Webanwendungen

Das HTTP-Protokoll ist ein *zustandsloses* Protokoll, dies beschreibt den Umstand, dass mehrere Anfragen an denselben Webserver prinzipiell unabhängig voneinander sind. Die Zustandslosigkeit des HTTP-Protokolls resultiert aus dem eingesetzten Request/Response-Verfahren: Ruft man etwa in einem Browser die URL `www.example.com/index.html` auf, so wird zunächst eine (TCP-)Verbindung zu Port 80 des Hosts `www.example.com` aufgebaut. Sobald die Verbindung hergestellt ist, fordert der Browser per HTTP-Request die Seite `index.html` an. Diese Anfrage wird vom Webserver des Rechners `www.example.com` verarbeitet und beantwortet (HTTP Response). Die Antwort enthält den Inhalt der HTML-Seite `index.html` (oder Angaben zu Ausnahmen/Fehlern). Sobald die Antwort vollständig übertragen wurde, wird die Netzwerkverbindung zwischen Server und Browser geschlossen. In `index.html` referenzierte Ressourcen (beispielsweise Bilder und Stylesheets) lösen ggf. neue, unabhängige Request/Response-Zyklen aus, jeweils mit einer eigenen Netzwerkverbindung.

Sobald eine Webseite vollständig übertragen wurde, besteht keine Verbindung mehr zwischen dem Webserver und dem Browser. In vielen Fällen ist es aber wünschenswert Nutzer/inne/n Änderungen auf dem Server sofort mitzuteilen, etwa den Inhalt einer Chat-Nachricht, Preisänderungen oder die Verfügbarkeit von Artikeln. Da der Webserver keinen Verbindungsaufbau zu einem Browser initiieren kann, muss dazu entweder eine offene Verbindung zwischen Browser und Server bestehen, oder der Browser muss regelmäßige Anfragen (via AJAX) an den Server senden, um ggf. vorhandene Neuigkeiten zu erhalten (Polling).

Die Nachteile von Polling liegen auf der Hand:

- viele unnötige Anfragen an den Server;
- kürzeste Latenzzeit abhängig vom Polling-Intervall.

Abgesehen von Polling existieren weitere Ansätze für Pseudo-Push-Technologien, insgesamt werden diese unter den Begriff „Comet“- oder „HTTP Streaming“-Technologien zusammengefasst. Ebenfalls verwendet werden die Begriffe „Ajax Push“ und „Reverse Ajax“ – diese sind aufgrund der expliziten Nennung einer Technologie jedoch als Sammelbegriffe weniger sinnvoll. Allen Comet-Technologien ist gemein, dass es sich entweder um Polling handelt, oder Tricks verwendet werden, um Verbindungen offen zu halten („long polling“).

WebSocket

Das WebSocket-Protokoll (W3C, 2012) beschreibt ein auf HTTP aufgesetztes Protokoll für persistente, bidirektionale Verbindungen zwischen Server und Client (Browser). Die meisten aktuellen Browser bieten bereits gute Unterstützung für das WebSocket-Protokoll – leider treten bei einigen Telekommunikationsanbietern noch Fehler beim Routing von WebSocket-Verbindungen auf, besonders bei Internetzugängen über Mobilfunknetze. Der Hintergrund ist, dass einige Mobilfunk-Internetanbieter die Kommunikationsinhalte zwischen Server und Empfänger automatisch manipulieren. Beispielsweise wird der Quelltext von HTML-Seiten umgeschrieben, um im HTML-Code referenzierte Bilder durch Versionen mit geringerer Auflösung zu ersetzen. Ein weiterer Grund ist, dass Verbindungen zwischen Telekommunikationsanbietern und mobilen Geräten ggf. eingefroren werden, d.h. ein Rechner beim Anbieter übernimmt dabei als Proxy die Rolle des Empfängers. Dies dient der Verbesserung von Akkulaufzeiten und erleichtert den Wechsel zwischen Funkzellen.

Bei der Verwendung von SSL-Verschlüsselung treten diese Probleme jedoch nicht auf, durch die Verschlüsselung können Router und Proxy Server den Inhalt der Kommunikation weder öffnen noch manipulieren.

Die größere Herausforderung beim Einsatz von WebSockets liegt jedoch darin, dass persistente bidirektionale Verbindungen ein ganz anderes Verarbeitungsmodell verlangen. „Normale“ Webanwendungen basieren auf dem Request/Response-Zyklus, d.h. jede Anfrage wird einzeln und unabhängig voneinander verarbeitet. Die meisten Web Applikation Frameworks sind entsprechend auf diese Art der Verarbeitung hin ausgerichtet. Häufig werden Push-Services daher als eigener Dienst eingerichtet und erfordern gesonderte Kommunikation zwischen Push-Server (auch Publication/Subscription Server oder kurz Pub/Sub-Server genannt) und Web-Applikations-Server.

Auf der Client-Seite, d.h. im Browsermacht sich die Verankerung des HTTP-Zyklus ebenso bemerkbar: WebSocket-Verbindungen existieren im Dokument-Kontext einer HTML-Seite, d.h. Entladen des Dokuments (beim Aufruf einer neuen Seite über einen Link) schließt eventuell geöffnete WebSocket-Verbindungen. Push-Nachrichten, die in der Zeit zwischen dem Entladen eines Dokuments und Laden des nächsten Dokuments gesendet werden, müssen somit auf dem Server gepuffert werden.

Nun könnte man alle internen Links so umgestalten, dass statt einem neuen HTML-Dokument lediglich die geänderten Daten über die bestehende WebSocket-Verbindung bezogen werden. Man sieht sich dann aber mit einer modernen Version des uralten Frameset-Problems konfrontiert: Alle Unterseiten zeigen dieselbe URL in der Adresszeile des Browsers, d.h. die Verwendung direkter Links und das Setzen von Bookmarks für tiefer liegende Inhalte werden erschwert.

Etherpad, Chat und Online-Awareness-Funktion in C3LLO

Push-Technologien sind besonders für die Entwicklung kollaborativer Werkzeuge und Anwendungen von Bedeutung. Eine nützliche Anwendung für E-Learning-Zwecke ist Etherpad (GitHub: Etherpad-Lite, 2014). Dabei handelt es sich um einen einfachen Texteditor, der die synchrone und asynchrone gemeinschaftliche Bearbeitung von Texten erlaubt. Etherpad ist eine kostenlose, eigenständige Webanwendung (Open-Source, Apache License v. 2.0) und bietet eine HTTP API, die es ermöglicht, Etherpad in beliebige Web-Anwendungen zu integrieren. Über die API ist es zudem möglich, Zugangsbeschränkungen für Nutzer- und Gruppen zu definieren; durch den Betrieb hinter einem Reverse Proxy kann die Kommunikation ohne großen Aufwand mit SSL verschlüsselt werden. Es sollte an dieser Stelle erwähnt werden, dass Etherpad aktuell auf Comet-Technologien zurückgreift, prinzipiell aber für die Verwendung von WebSocket konzipiert ist.⁴

Als eigene Entwicklungen nutzen die Chat-Funktion und die Online-Awareness-Funktion in C3LLO das WebSocket-Protokoll. Als WebSocket-Integration für Ruby on Rails wird die Bibliothek ‚websocket-rails‘ verwendet (Knox, 2014). Über diese Bibliothek können die WebSocket-Funktionen bequem direkt in der Rails-Anwendung verwendet werden. Im vorigen Kapitel wurde kurz erwähnt, dass WebSocket-Verbindungen beim Wechsel zwischen Seiten durch normale Hyperlinks geschlossen werden. Dieser Umstand äußert sich vor allem in der Echtzeit-Darstellung einer ‚Wer ist Online?‘-Liste und würde ohne Behandlung dazu führen, dass Nutzer/innen beim Seitenwechsel kurzzeitig in den Offline-Status wechseln. Dies wird vermieden, indem das Ereignis ‚Nutzer X hat die Verbindung geschlossen‘ einige Sekunden verspätet und nur dann ausgelöst wird, falls innerhalb der Pufferzeit kein neuer Verbindungsaufbau stattfindet.

3.4 Pilotierung und Nutzungsstatistiken

C3LLO ist seit September 2013 in Verwendung, die ersten Nutzer/innen sind die Teilnehmenden des Zertifikatprogramms ‚Weiterbildendes Studium Windenergietechnik und -management‘ von ForWind⁵. Im Wintersemester 2013/2014 wurden Pilotmodule in den berufsbegleitenden Studiengängen der Carl von Ossietzky Universität Oldenburg durchgeführt. Ab dem Sommersemester 2014 wird C3LLO flächendeckend für alle Module der

4 <https://github.com/ether/etherpad-lite/issues/686> (09.07.2014).

5 www.forwind.de/ (09.07.2014).

berufsbegleitenden Studiengänge eingesetzt und bisher von rund eintausend registrierten Studierenden verwendet.

Die Pilotierungsphase wurde so früh wie möglich durchgeführt, um Anregungen und Verbesserungsvorschläge der Nutzenden noch vor Abschluss der Entwicklung berücksichtigen zu können. Die zum Ende der Pilotierungsphase durchgeführte qualitative Evaluation ergab insgesamt eine hohe Zufriedenheit mit der neuen Lernumgebung sowie eine Reihe von Anregungen und Verbesserungsmöglichkeiten. Die Mehrheit der Anregungen konnte ad hoc als Verbesserungen von C3LLO umgesetzt werden, während ein kleinerer Teil in die Diskussion und Planung der Weiterentwicklung eingeflossen ist.

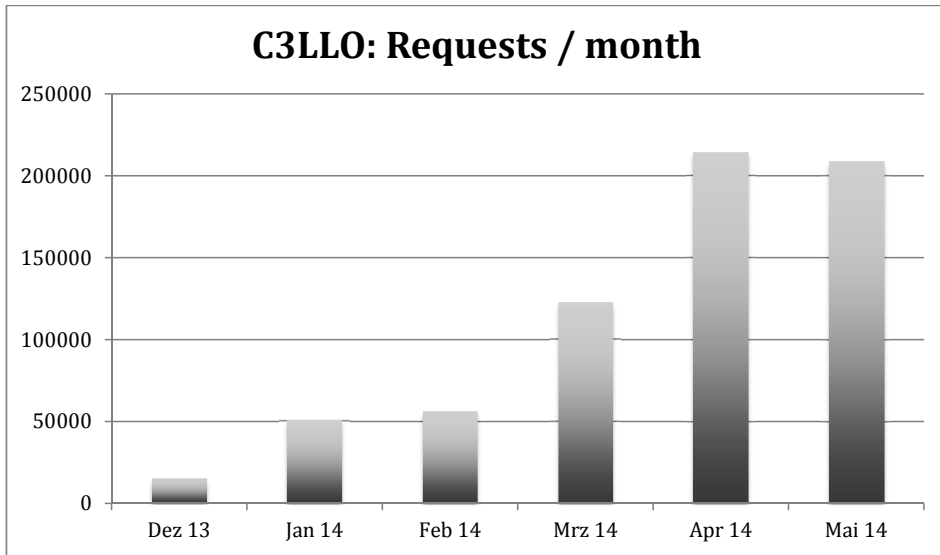


Abbildung 4: Vom Applikations-Server verarbeitete Anfragen pro Monat (eigene Darstellung)

Abbildung 4 zeigt die zeitliche Entwicklung der Anzahl der verarbeiteten Anfragen pro Monat. Die dargestellten Daten betreffen Anfragen an den Rails-Applikations-Server und entsprechen der Anzahl angeforderter Seiten. Eine Seite beinhaltet typischerweise 20 bis 40 statische Ressourcen (Bilder, Stylesheets, Skripte), Ausnahmen bilden Anfragen via AJAX – diese enthalten oftmals keine statischen Ressourcen. Als grobe Abschätzung für die Auslastung des Webserver müssen die dargestellten Zahlen um eine Größenordnung erhöht werden.

In Abbildung 5 ist die gemittelte tägliche Nutzung nach der Tageszeit dargestellt. Es handelt sich dabei ebenfalls um Anfragen an den Rails-Applikations-Server (s.o.).

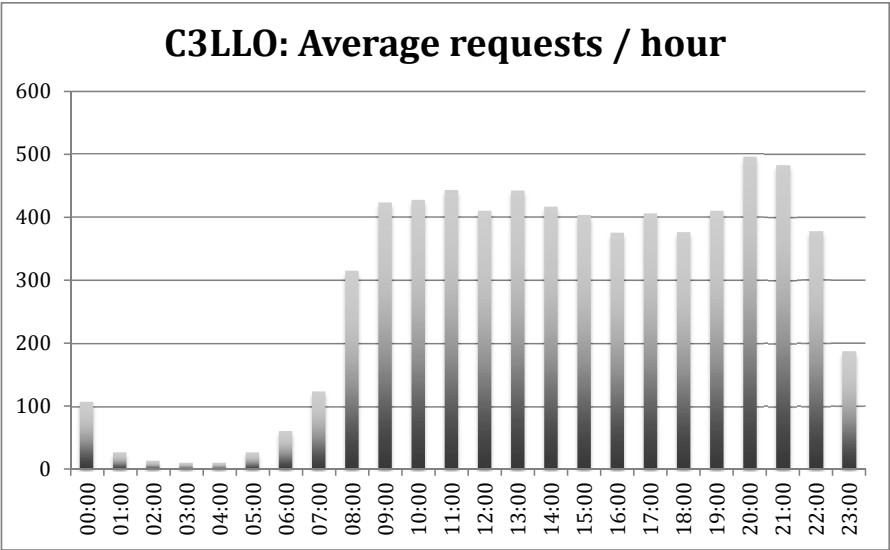


Abbildung 5: Durchschnittliche Zahl der Anfragen nach Tageszeit im Mai 2014 (eigene Darstellung)

3.5 Entwicklungsaufwand

Die Entwicklung von C3LLO begann im Frühjahr 2012, während der erste Praxiseinsatz der Plattform im Herbst 2013 stattgefunden hat. Tabelle1 zeigt eine Übersicht der in diesem Zeitraum aufgewendeten personellen Ressourcen. Zusammen ergibt sich ein personeller Aufwand von 13 Vollzeit-Monaten. Dabei ist festzuhalten, dass alle beteiligten Entwickler im Vorfeld auf langjährige Erfahrung im Bereich der Webentwicklung mit einschlägiger Erfahrung in der Entwicklung von Lernplattformen zurückblicken konnten.

Tabelle1: Personeller Entwicklungsaufwand der Lernumgebung C3LLO (eigene Darstellung)

Arbeitsgebiet	Monate	%-beschäftigt
Programmierung	16	50
Programmierung	2	100
Design	3	100

Dabei muss beachtet werden, dass der Zustand der Plattform im Herbst 2013 noch nicht als ‚fertig‘ bezeichnet werden kann. Der Aufwand für die Neuentwicklung einer voll einsatzfähigen Lernplattform muss im Minimum mit 18 Vollzeit-Monaten veranschlagt werden. Der Umfang der Quelltexte liegt aktuell bei 110.000 Zeilen (inklusive HTML Temp-

lates, CSS & Javascript). Seit Beginn der Entwicklung wurden 1.193 Änderungen („Commits“) in das Git-Repository⁶ eingestellt.

4 Fazit

Die vom C3L betreuten berufsbegleitenden Studiengänge sind speziell auf Personen mit eingeschränktem Zeitbudget ausgerichtet und wenden ein typisches Blended-Learning-Konzept mit alternierenden Online- und Präsenzphasen an. Die für die Onlinephasen benötigte Lernumgebung ist in untergeordneter Weise dem Instruktionsdesign anzupassen, das auf einem konstruktivistisch, kompetenzorientierten didaktischen Modell fußt, bei dem die Qualität der Moderation maßgeblich ist für die Qualität der Lernprozesse. Die zentrale Aufgabe der C3L-Lernumgebung ‚C3LLO‘ besteht demnach in der bestmöglichen Unterstützung des Moderationsprozesses durch Werkzeuge für synchrone und asynchrone Kommunikation. Die Lernumgebung leistet nicht nur eine Reduktion auf die tatsächlich benötigten Funktionen und ermöglicht hierdurch eine intuitiv einfache Nutzung, sondern erleichtert auch eine direkte intra-institutionelle Kommunikation zwischen didaktischem Personal und Entwicklern der Lernumgebung. Gerade letztgenannter Punkt ist nicht zu unterschätzen, da sich hieraus oftmals Schwierigkeiten in der Kommunikation ergeben. Die größere strukturell-organisatorische und persönliche Nähe erleichtern die Kommunikation zwischen Expert/inn/en für Didaktik und IT in erheblichem Maße.

Im Zuge der in den letzten beiden Jahren erfolgten Neu-Implementierung von C3LLO auf Basis von Ruby on Rails zeigte sich, dass die damit verbundenen Herausforderungen auch mit einem kleinem Entwicklerteam und begrenzten Mitteln erfolgreich bewältigt werden können. Hervorzuheben ist dabei, dass Ruby on Rails von den Entwicklern als eine, für das Vorhaben sehr gute technische Basis empfunden wurde. Die leicht verständliche Struktur und die Konventionen von RoR unterstützen die Erstellung von übersichtlichem, leicht zu wartendem Programmcode.

C3LLO bildet die organisatorischen Strukturen der berufsbegleitenden Studiengänge auf eine intuitive, übersichtliche Weise ab. Neue Inhalte sind für die Nutzer/innen schnell erkenn- und auffindbar. Das Instruktionsdesign wird durch die Kommunikationsfunktionen gezielt unterstützt, wobei der Schwerpunkt der Kommunikation bewusst auf den asynchronen Modus gelegt wurde, um den heterogenen zeitlichen Einschränkungen der Zielgruppe Rechnung zu tragen.

Literatur

Dahlstrom, E. J. (2013). *ECAR Study of Undergraduate Students and Information Technology*. In E. C. Research (Hrsg.). Louisville.

6 <http://git-scm.com/> (09.07.2014).

- Fette, I. & Melnikov, A. (2011). *The WebSocket Protocol*. IETF (Hrsg.), URL: <http://tools.ietf.org/html/rfc6455> (27.05.2014).
- GitHub (2014). *Etherpad-Lite*. URL: <https://github.com/ether/etherpad-lite> (29.05.2014).
- Hansson, D. H. (ohne Datum). *Ruby on Rails*. URL: <http://rubyonrails.org/> (02.06.2014).
- Kerres, M. (2013). *Mediendidaktik: Konzeption und Entwicklung mediengestützter Lernangebote, 4 Ausgabe*. München: Oldenbourg.
- Knox, D. (2014). *GitHub: websocket-rails*. URL: <https://github.com/websocket-rails/websocket-rails> (29.05.2014).
- Ruby-Community (ohne Datum). *Die Programmiersprache Ruby*. URL: <https://www.ruby-lang.org/de/> (02.06.2014).
- Schulmeister, R. (2005). Kriterien didaktischer Qualität im E-Learning zur Sicherung der Akzeptanz und Nachhaltigkeit. In D. Euler & S. Seufert (Hrsg.), *E-Learning in Hochschulen und Bildungszentren* (S. 473-492). München: Oldenbourg.
- W3C. (2012). *XMLHttpRequest Level 2*. URL: <http://www.w3.org/TR/2012/WD-XMLHttpRequest-20120117/> (27.05.2014).
- W3C. (2012). *The WebSocket API*. URL: <http://www.w3.org/TR/2012/CR-websockets-20120920/> (27.05.2014).
- W3C. (2013). *File API*. URL: <http://www.w3.org/TR/2013/WD-FileAPI-20130912/> (23.05.2014).
- W3C. (2014). *HTML5*. URL: <http://www.w3.org/TR/2014/CR-html5-20140429/> (27.05.2014).
- Zawacki-Richter, O. (2013). Instruktionsdesign für berufstätige Studierende. In A. Hanft & K. Brinkmann (Hrsg.), *Offene Hochschulen Die Neuausrichtung der Hochschulen auf lebenslanges Lernen* (S. 199-205). Münster: Waxmann.
- Zawacki-Richter, O., Hohlfeld, G. & Müskens, W. (2014). Mediennutzung im Studium. In: A. Hanft, H. Rübken, O. Zawacki-Richter & R. Götter. *Schriftenreihe zum Bildungs- und Wissensmanagement* (1).